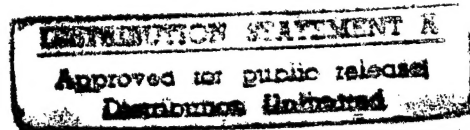**DATE:** 4/01/97

**CONTROLLING OFFICE FOR THIS DOCUMENT IS:**
DIRECTOR, Army High Performance Computing
Research Center (AHPCRC)
Army Research Laboratory
Aberdeen, MD

**POC:** Director (Tayn E. Tezduyar)

**DISTRIBUTION STATEMENT A:** Public release

# Using High Performance Computers in Distributed Interactive Visualization Applications at CEWES

*Fall 95*

by Michael Stephens (AHPCRC-MSCI)

Traditionally, when one considers how visualizations are created in the realm of high performance computing (HPC), the most common model has two distinct phases. Phase one, in which the data is generated and where the role of the supercomputer is simply that of a data firehose spewing forth oceans of data. In phase two, this usually massive amount of data is then transferred to a graphics workstation where an appropriate visualization application is run to produce renderings that will allow for the exploration of the information. Often there are two major problems encountered in this model. First, the data must usually be limited to accommodate the smaller workstation memories (RAM and disk). Second, in many cases the time needed to compute the visual images is unworkably long, making interaction with the visualization impossible. In the model described above the supercomputer is not part of the visualization application at all.

Recent technological developments in high-speed networks and communication protocols have served to bring high performance computers out of the shadows and into the spotlight with the other components of visualization applications. Tapping the computational power of supercomputers for visualization algorithms and exploiting the high-speed network's ability to move large chunks of data rapidly to and from high performance graphics workstations enables the creation of an environment where distributive, interactive and collaborative visualization applications are possible. A very exciting extension of this environment is to connect HPC resources together to facilitate the notion of generating "on the fly'' data fed to the visualization application. This hints of a long sought after goal in HPC- that of the user being able to interactively steer the calculation.

Presented below are two visualization applications demonstrating the integrated use of the three major components in HPC: supercomputers, high-speed networks, and high performance graphics. These applications are primarily the work of John West and Alex Carrillo both of the Department of Defense (DoD)'s HPC Center at the Corps of Engineers Waterways Experiment Station (CEWES) in collaboration with subject area researchers, Stacy Howington (Hydraulics Lab/CEWES) and John Peters and David Horner (Geotechnical Lab/CEWES).

The first application described below allows the interactive, collaborative exploration of large precomputed time-dependent

datasets. The supercomputer "digests'' these datasets into a
graphical form and in turn communicates this image to the
graphics workstations. The second application utilizes HPC
resources to compute "live'' data which is then processed by
additional HPC resources to produce the visualization. The key
features in both applications are: 1) a user controlled, highly
interactive closed loop and 2) the use of a supercomputer as an
integral part of the visualization application.

## Immiscible Flow Visualization

CEWES has responsibility within the DoD for assisting in the
clean-up of contaminated groundwater on military installation.
Therefore, there is a strong research effort dedicated to
understanding the nature of the flow of contaminants and water in
various soil types. A common source of contamination is from
leaking underground fuel tanks. Being able to predict the bulk
flow behavior of multiple fluid phases in underground conditions
is absolutely necessary in formulating effective and efficient
clean-up strategies. Such predictions are the result of large-
scale numerical computations for the solution of the fundamental
equations describing the physics of the situation. These large-
scale computations, however, depend on knowledge of physical
parameters such as viscosities, surface tensions, contact angles
and absolute and relative permeabilities. Such physical
parameters are measured in expensive and time consuming
laboratory experiments which deal with small porous=20samples. To
aid in the interpretation of such experimental results for
the constitutive relations used in the large-scale computations,
models of the pore size process are required. For this reason,
CEWES researchers are investigating pore scale models to provide
insight into the displacement of a wetting fluid such as water by
an immiscible contaminant fluid like an organic solvent.

One approach being utilized in this effort is the lattice
Boltzmann permeameter. This is a conceptually different way of
solving for the fluid motion from the more traditional numerical
schemes such as finite difference or finite element methods used
to solve the governing Navier-Stokes equations for incompressible
flow.

In the lattice Boltzmann approach, the governing physical laws
are based on the Boltzmann transport equations - a set of kinetic
rules that describe the motion of particles and their associated
mass and momentum in a lattice. It is the time rate of change of
the particle distribution function which is obtained.
The rate of change is the number of fluid particles entering a
state minus the number of particles leaving that state. It has
been shown that this formulation is equivalent to the Navier-
Stokes description.

This novel approach was first developed by Los Alamos National
Laboratory's geoanalysis group in a joint project with Mobil Oil
Corporation to help determine the economic viability of various

oil field reservoirs. Cast in these terms the solution to this pore scale problem can be obtained by employing extremely fast and efficient parallel software techniques. To resolve the flow in the pores significant detail in the discretization simulations are made with between one million to ten million lattice sites. These calculations are performed on the AHPCRC-CM-5.

The visualization application follows the client-server programming model. The server, which is the most computationally demanding component, is a highly optimized volume rendering engine embodying a ray casting algorithm. Ray casting is a technique in which rays are shot from every pixel in an image plane into the data. The pixel value-pixel color and opacity-is determined by accumulating the contributions along the ray as it passes through the data. The images which result from this look as if the data were made of different colored Jello. Since rays do not interact with one another this process is highly vectorizable. Currently this code runs on the DoD's CRAY C916 at CEWES.

The client side of this code controls the image plane size and orientation (view direction), color and opacity assigned to the various constituents. These user parameters are sent to the server for the calculation of the view. Once the image is computed it is sent to the client for display. In a typical operation the user interactively navigates through the dataset looking for meaningful structures. At CEWES this client runs on an SGI workstation, however, since it does not make use of any special graphics hardware, the client can be run on any workstation running X windows.

The server typically delivers 21 frames per second of a 400 x 400 image while processing transient data sets which are 201 x  201 x 201 samples.  This represents an approximate performance of 300 MegaFLOPS.

Figure 1 shows an example frame. In this picture the solid soil particles are shown in red, the displacing fluid (organic phase) is shown in green while the incumbent fluid (water) is rendered as totally transparent.  Notice the flow "fingers'' of the organic phase. The ability to change colors and opacities of the three phases allows the researcher navigating the data to minimize or completely turn off any of the phases thereby rendering internal fluid interface structure.

This application supports multi-clients allowing two or more researchers to collaborate by viewing the same data set simultaneously.

## Interactive Particle Model

The second application centers on a project from the Soil Particle Model program in the Mobility Systems Division in the Geotechnical Laboratory at CEWES. In this effort researchers are

using particle theory techniques to describe soil behavior during
large strain, discontinuous deformations. Such deformations occur
in vehicle sinkage and plowing problems as well as in the
interaction of tires and tracks in soft soils. The particle
theory description of the soil response offers an attractive
alternative to the classical continuum mechanics formulation.
Due to compatability relationships, the discrete particle model
does not limit the analysis of this class of problems
to the case of small strains as it does in the continuum
approach. The particle model treats the soil as a collection of
individual unconnected particles whose motion are controlled by
Newton's law of momentum conservation. This more general
computational procedure extends the Army's capability to model
vehicle-soil interaction problems.  As an added benefit, this
work helps enhance the Army's ability to model a wide variety of
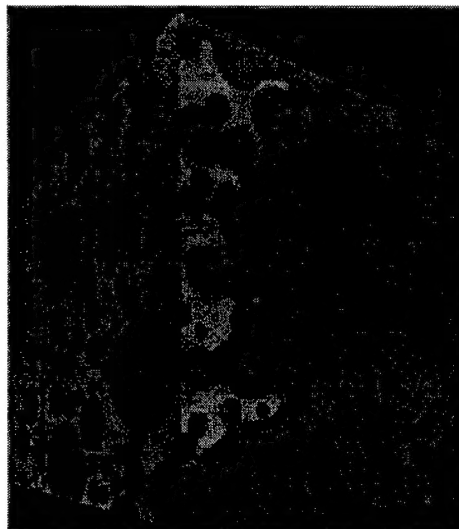soil-interface problems.



Figure 1. Contaminant flow around soil particles

The design of this visualization application centers on
classifying the tasks to be performed into three main processes:
modeling the physics, data reduction, and data visualization.
Each of the three processes communicates with the other two, with
the bulk of the communication being between the physical modeling
and the data reduction processes.

The physical modeling process is most efficiently accomplished by
running the particle interaction model code on a parallel,
distributed memory system. The result of this model is the
position and average stress of the myriad particles. The data
reduction process is the key to the "real time'' and remote
visualization capabilities of the application. Using a high-speed
connection, the large volume of particle data is continually
passed from the particle model to the data reducer. Using a
viewing transformation obtained from the remote user, the data
reducer determines which particles are visible. The visibility

culling uses a technique much like the ray casting scheme
discussed in the first application. While the physical model
typically requires from tens to hundreds of thousands of closely
packed particles to adequately model the physical situation, only
a small fraction of these particles are visible. Therefore, the
communication burden between the remote user and the data
reducer as well as the computational requirements on the remote
workstation are significantly reduced.

The graphics workstation renders the visible particles, supplies
the user viewing interaction and allows the user to manipulate a
prescribed solid object in the particle model. For example,
moving a cylinder or cone object in a particle mass is an
important step in understanding track-soil interactions. In
this way the user not only influences what is seen but also
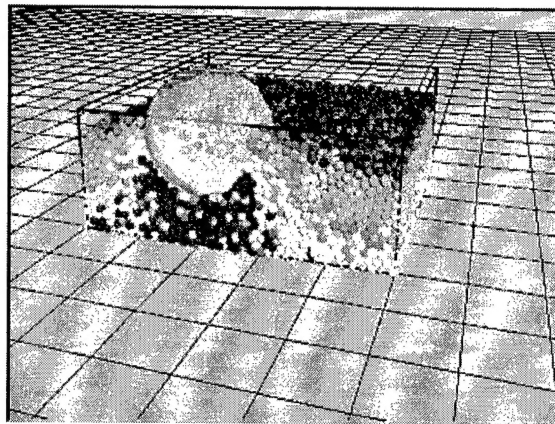guides the physical modeling process.



Figure 2. Particle dynamics simulation.

Figure 2 shows a frame from just such an interaction. Here a
sphere is being controlled by the user and the smaller particles
react to the sphere's motion. The color indicates the average
pressure on the particle, red being high values and blue being
low values.